[19]中华人民共和国国家知识产权局

[51]Int. Cl[7]
G06F 9/48
H04L 9/00

[12]发明专利申请公开说明书

[21]申请号 01103000.3

[43]公开日 2001年8月22日

[11]公开号 CN 1309351A

[22]申请日 2001.2.14 [21]申请号 01103000.3

[30]优先权

[32]2000.2.14 [33]JP [31]035898/2000

[32]2000.5.8 [33]JP [31]135010/2000

[71]申请人 株式会社东芝

地址 日本神奈川县

[72]发明人 桥本干生 寺本圭一 齐藤健
白川健治 藤本谦作

[74]专利代理机构 中国国际贸易促进委员会专利商标事务所

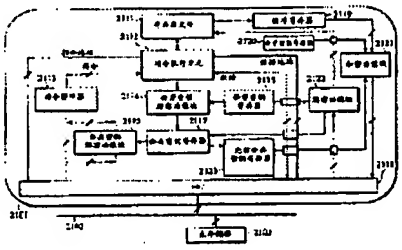
代理人 吴丽丽

权利要求书4页 说明书43页 附图页数15页

[54]发明名称 抗干预微处理器

[57]摘要

在多任务环境下，抗干预微处理器保存一个其执行被中断的程序 的上下文信息，其中该上下文信息含有指明该程序的执行状态和该程 序的执行码密钥的信息。通过从保存的上下文信息恢复该程序的执行 状态，可以重新启动该程序的执行。利用微处理器的公开密钥可以将 此上下文信息加密，然后利用微处理器的秘密密钥进行解密。



ISSN1008-4274

知识产权出版社出版

01.02.14

权 利 要 求 书

1. 一种具有不能被读出到外部的唯一秘密密钥和与该唯一秘密密钥对应的唯一公开密钥的微处理器，该微处理器包括：

读取单元，被进行配置以从外部存储器读出多个利用不同执行码密钥加密的程序；

解密单元，被进行配置以利用各自解密密钥，对多个通过读取单元读出的程序进行解密；

执行单元，被进行配置以执行多个利用解密单元解密的程序；

上下文信息保存单元，被进行配置以将其执行被中断的一个程序的上下文信息保存到外部存储器或保存到在微处理器内部设置的上下文信息存储器，该上下文信息含有指明此程序的执行状态和此程序的执行码密钥的信息；以及

重新启动单元，被进行配置以通过从外部存储器或上下文信息存储器读出上下文信息并通过从上下文信息中恢复此程序的执行状态，重新启动执行此程序。

2. 根据权利要求 1 所述的微处理器，其中所配置的上下文信息保存单元利用公开密钥对上下文信息进行加密，并将加密上下文信息保存到外部存储器；以及

所配置的重新启动单元通过从外部存储器读出加密上下文信息，利用秘密密钥解密加密上下文信息，以及从解密上下文信息中恢复一个程序的执行状态，重新启动此程序的执行。

3. 根据权利要求 2 所述的微处理器，其中仅当包含在解密上下文信息内的解密执行码密钥与此程序的执行码密钥一致时，重新启动单元才重新启动此程序的执行。

4. 根据权利要求 2 所述的微处理器，其中重新启动单元将包含在解密上下文信息内的解密执行码密钥用作解密密钥以解密此程序。

5. 根据权利要求 1 所述的微处理器，其中所配置的上下文信息保存单元以明文形式将上下文信息保存到此程序被中断后所执行的另一

01-02-14

个程序不可读的上下文信息存储器；以及

通过从上下文信息存储器读出上下文信息并从上下文信息恢复此程序的执行码，所配置的重新启动单元重新启动此程序的执行。

6. 根据权利要求 5 所述的微处理器，其中重新启动单元根据另一个程序规定的指令重新启动此程序的执行。

7. 根据权利要求 6 所述的微处理器，其中在此程序的执行被中断时，上下文信息保存单元将上下文信息保存到上下文信息存储器，并利用公开密钥将上下文信息存储器内的上下文信息加密，然后根据另一个程序规定的另一条指令的执行，将加密上下文信息存储到外部存储器。

8. 根据权利要求 5 所述的微处理器，其中在此程序的执行被中断时，上下文信息保存单元将上下文信息保存到上下文信息存储器，利用公开密钥将上下文信息存储器内的上下文信息加密，然后将加密上下文信息存储到另一个程序规定的外部存储器。

9. 根据权利要求 1 所述的微处理器，其中所配置的上下文信息保存单元产生作为临时密钥的随机数、加密上下文信息、然后将加密上下文信息存储到外部存储器，加密上下文信息含有：第一数值，通过对信息进行加密获得，利用临时密钥指明此程序的执行状态；以及第二数值，通过利用公开密钥加密临时密钥获得；以及

通过从外部存储器读出加密上下文信息，利用秘密密钥由包含在加密上下文信息内的第二数值解密获得临时密钥，利用解密的临时密钥由包含在加密上下文信息内第一数值解密出指明执行状态的信息，以及从解密上下文信息恢复此程序的执行状态，所配置的重新启动单元重新启动此程序的执行。

10. 根据权利要求 9 所述的微处理器，其中上下文信息保存单元保存还含有利用此程序的执行码密钥对临时密钥进行加密获得的第三数值的加密上下文信息。

11. 根据权利要求 10 所述的微处理器，其中重新启动单元利用秘密密钥由包含在加密上下文信息内的第二数值解密获得第一临时密

钥，并利用第一解密临时密钥由包含在加密上下文信息内的第一数值解密获得指明执行状态的信息，同时利用该程序的执行码密钥由包含在加密上下文信息内的第三数值解密获得第二临时密钥，然后只在第一解密的临时密钥与第二解密的临时密钥一致时，重新启动此程序的执行。

12. 根据权利要求 1 所述的微处理器，该微处理器进一步包括：

执行状态存储单元，用于存储当前执行程序的执行状态；以及

执行状态初始化单元，被进行配置以在此程序被中断后而在另一个程序开始之前，将执行状态存储单元的内容初始化为规定数值或将执行状态存储单元的内容加密。

13. 根据权利要求 1 所述的微处理器，该微处理器进一步包括：

密钥读取单元，被进行配置以从外部存储器读出被事先利用公开密钥加密的各程序的执行码密钥；以及

密钥解密单元，被进行配置以利用秘密密钥解密通过密钥读取单元读出的执行码密钥；

其中解密单元利用作为解密密钥的执行码密钥解密各程序。

14. 根据权利要求 1 所述的微处理器，该微处理器进一步包括：

执行状态存储单元，用于存储当前执行程序的执行状态和将被当前执行程序处理的数据的加密属性；以及

数据加密单元，被进行配置以根据存储在执行状态存储单元的加密属性对将由当前执行程序处理的数据进行加密。

15. 根据权利要求 1 所述的微处理器，该微处理器进一步包括：

执行状态存储单元，用于存储当前执行程序的执行状态、将被当前执行程序处理的数据的加密属性以及用于规定加密属性的加密属性规定信息；

相关信息写入单元，被进行配置以将涉及加密属性规定信息并含有利用秘密密钥获得的签名的相关信息写入外部存储器；

相关信息读出单元，被进行配置以根据将由当前执行程序引用的数据的地址从外部存储器读出相关信息；

01.02.14

数据引用许可单元，被进行配置以利用公开密钥验证包含在相关信息内的签名，并且只有当相关信息内的签名与微处理器的原始签名一致时，才允许当前执行程序根据相关信息和规定信息的加密属性，通过确定密钥和用于数据引用的算法进行数据引用；以及

数据加密单元，被进行配置以根据存储在执行状态存储单元的加密属性将由当前执行程序引用的数据加密。

16. 根据权利要求 1 所述的微处理器，该微处理器进一步包括：

高速缓冲存储器，用于以高速缓存行为单位高速缓存多个程序的明文指令和明文数据，该高速缓冲存储器具有属性区用于各高速缓存行，指明在解密其指令被高速缓存到各高速缓存行的各程序或其执行会将明文数据高速缓存到各高速缓存行的各程序时用于唯一标识解密密钥的解密密钥标识符；

高速缓存访问控制单元，被进行配置以只有当加密属性对一个高速缓存行指明的解密密钥标识符与加密属性对另一个高速缓存行指明的解密密钥标识符一致时，允许通过根据另一个高速缓存行内的一个高速缓存数据执行一个存储在一个高速缓存行的高速缓存程序引起的数据引用。

17. 根据权利要求 16 所述的微处理器，其中当不允许进行数据引用时，将新数据从外部存储器高速缓存到另一个高速缓存行。

18. 根据权利要求 16 所述的微处理器，其中当不允许进行数据引用时，保护异常中断此高速缓存程序的执行。

19. 根据权利要求 1 所述的微处理器，其中执行单元还执行明文程序，并具有调试功能块，在明文程序的执行期间，当执行特定地址或地址区域的程序时或将数据引用到特定地址或地址区域的数据时，该调试功能块用于产生异常，在执行加密程序期间，此调试程序无效。

20. 根据权利要求 1 所述的微处理器，其中该微处理器的各组成单元包含在单一芯片或单一封装内。

01.02.14

说　　明　　书

抗干预微处理器

本发明涉及可以在多任务程序执行环境下防止非法变更执行码和非法处理目标数据的微处理器。

最近几年，微处理器的性能得到显著改善，以致微处理器除了具有传统的诸如计算和图形功能外，还可以实现对视频图像和音频声音的再生和编辑。通过在为最终用户设计的系统（以下简称：PC）中实现这种微处理器，用户可以在监视器上欣赏各种视频图像和音频声音。此外，通过将 PC 的再生视频图像和音频声音的功能与计算能力相结合，可以改善对游戏等的适用性。这种微处理器不是专为某种特定硬件设计的而是可以在各种硬件中实现，因此其优势在于，通过简单更换执行程序的微处理器，已经拥有 PC 的用户花费不多就可以欣赏视频图像和音频声音的再生和编辑。

如果在 PC 上处理视频图像和音频声音，就会产生原始图像和音乐的版权保护问题。在 MD 或数字视频重放装置中，通过在这些装置中事先实现防止非法复制的机制，可以防止无限复制。虽然这种装置还在制造，但是试图通过拆除或改变这些装置来进行非法复制的情况却很少，而且世界范围内的趋势是通过法律禁止制造和销售为了进行非法复制能够改变的装置。因此，由于基于硬件进行非法复制造成的损害并不很严重。

然而，在 PC 上对图像数据和音乐数据进行处理是通过软件进行的而不是通过硬件进行的，并且最终用户可以在 PC 上随意改变软件。即，如果用户具有某些知识，则通过分析程序并重写可执行软件，可以非常容易地进行非法复制。此外，不同于硬件的问题是，这样产生的用于非法复制的软件可以通过诸如网络的各种媒体迅速传播。

为了解决这些问题，用于再生诸如商业电影或音乐的版权保护内容的 PC 软件，传统上采用一种通过对软件进行加密防止分析和变更

01.02.14

的技术。这种技术就是抗干预软件（参考 David Aucsmith 等人在 Proceedings of the 1996 Intel Software Developer's Conference 上发表的“Tamper Resistant Software: An Implementation”）。

在防止通过 PC 向用户提供的有价值信息（不仅包括视频数据和音频数据而且包括文本和技术诀窍）的非法复制方面，以及在防止 PC 软件本身的技术诀窍被分析方面，抗干预软件技术仍然有效。

然而，抗干预软件技术是一种，通过在开始执行程序之前对要求保护的程序的一部分进行加密，在执行该部分之前对该部分立即进行解密并在该部分执行完毕后立即对该部分再加密，使得难于利用诸如反汇编程序或调试程序的软件工具进行分析。因此，只要处理器可以执行该程序，通过从程序的启动处开始一步一步进行分析总可以对程序进行分析。

此事实成为版权所有人向系统提供版权保护内容用于利用 PC 再生视频数据和音频数据的障碍。

在这方面，其它抗干预软件应用程序也易受攻击，并且此事实成为通过 PC 进行高级信息服务和将含有企业或个人技术诀窍的程序应用到 PC 的障碍。

总之，在软件保护方面同样存在这些问题，此外，PC 是开放式平台，因此存在通过变更被确定为系统软件配置基础的操作系统（OS）进行攻击问题。换句话说，通过使用属于 OS 的特权，怀有恶意的熟练用户可以变更其自有 PC 的 OS 来废除或分析插入到应用程序内的版权保护机制。

当前的 OS 通过利用对存储器的特权操作功能和 CPU 中提供的特权执行控制功能，在计算机的控制下进行资源管理和资源使用仲裁。管理的目标包括传统目标（例如：设备、CPU 和存储资源）以及网络层或应用层 QoS（服务质量）。尽管如此，资源管理的基础仍然是对执行程序所需的资源进行配置。换句话说，分配 CPU 时间来执行此程序并将分配执行程序所需的存储空间是资源管理的基础。通过控制实现访问这些资源的程序的执行（通过分配 CPU 的时间和存储空间），对

01.02.14

其它设备、网络和应用层服务质量 Qos 进行控制.

OS 具有执行 CPU 时间分配和存储空间分配的特权. 换句话说, 为了对 CPU 分配时间, OS 具有在任意时间中断并重新启动应用程序的特权并具有在任意时间将分配到应用程序的存储空间的内容转移到不同分层的存储空间的特权. (通常) 通过利用应用程序的不同访问速度和访问能力隐匿分层存储系统, 将分配到应用程序的存储空间的内容转移到不同分层的存储空间的特权还用于为应用程序提供平面存储器空间.

使用这两种特权, OS 可以中断应用程序的执行状态并在任意时间对它进行快速转储, 并且在对它进行拷贝或重写之后重新启动它. 此功能还可以被用作分析隐藏在应用程序内的秘密的工具.

为了在计算机上防止应用程序被分析, 有几种对程序或数据进行加密的公知技术(例如: Hampson, 第 4,847,902 号美国专利、Hartman, 第 5,224,166 号美国专利、Davis, 第 5,806,706 号美国专利、Takahashi 等, 第 5,825,878 号美国专利、Buer 等人, 第 6,003,117 号美国专利、第 11-282667 号日本公开专利申请 (1999)). 然而, 这些公知的技术均未涉及防止程序运行过程和数据秘密被 OS 进行上述特权操作问题.

基于 Intel 公司开发的 X86 结构的传统技术 (Hartman, 第 5,224,166 号美国专利) 是一种通过利用规定的密钥 Kx 对执行码和数据进行加密以存储执行码和执行数据的技术. 密钥 Kx 可以被表示为 $E_{kp}[Kx]$ 的形式, 利用与嵌入到处理器内的秘密密钥 Ks 对应的公开密钥 Kp, 可以对 $E_{kp}[Kx]$进行加密. 因此, 只有知道 Ks 的处理器可以对存储器上的加密执行码进行解密. 将密钥 Kx 存储到处理器内被称为段式寄存器的寄存器.

利用这种机制, 通过对代码进行加密在某种程度上可以避免用户发现程序代码的秘密. 此外, 对于不知道代码密钥 Kx 的人来说, 由于密码原因难于根据其内涵或利用密钥 Kx 解密时可执行的新产生代码来变更代码.

然而, 采用这种技术的系统的缺点在于, 利用被称为上下文切换

的 OS 特权有可能对程序进行分析，而无需对加密的执行码进行解密。

更具体地说，当利用中断停止执行程序或当预期系统调用程序自行调用软件中断命令时，为了执行其它程序，OS 进行上下文切换。上下文切换操作将指明该点寄存器值的集合的程序执行状态（以下简称为：上下文信息）存储到存储器，并将事先存储到存储器的另一个程序的上下文信息再存入寄存器。

图 15 示出在 x86 处理器中使用的传统上下文存储格式。这里存储了应用程序使用的寄存器的所有内容。当再启动被中断的程序时，将该程序的上下文信息再存入寄存器。为了并行运行多个程序，上下文切换是不可缺少的功能。在传统技术中，在上下文切换时，OS 可以读